CHAPTER 3
PROGRAMMING

3.1 INTRODUCTION

This chapter provides basic information for programming the DZ11. A description of each DZ11 register, its format, programming constraints, and bit functions are presented to aid programming and maintenance efforts. Special programming features are also presented in this chapter.

3.1.1 Device and Vector Address Assignments

The DZ11's device and vector addresses are selected from the floating vector and device address space.

NOTE

The device floating address space is 160010_8 to 163776_8. The vector floating address space is 300_8 to 776_8.

Its floating address space follows the DJ11, DH11, DQ11, DU11, DUP11, LK11, and DMC11.

Its floating vector space follows the DC11; KL11/DL11-A, -B; DP11, DM11-A; DN11; DM11-BB and other modem control vectors; DR11-A; DR11-C; PA611 reader, PA611 punch; DT11; DX11; DL11-C, -D, -E; DJ11; DH11; GT40; LPS11; DQ11; KW11-W; DU11; DUP11; DV11; LK11-A; DWUN; and DMC11. If a DZ11 is installed in a system with any of the above listed options, then its assigned vector and device address should follow the vector and device address of the other options.

Two examples follow. First, the simplest case where there is only one DZ11.

Option	Address	Vector	Comment
GAP	160010		No DJ11s
GAP	160020		No DH11s
GAP	160030		No DQ11s
GAP	160040		No DU11s
GAP	160050		No DUP11s
GAP	160060		No LK11s
GAP	160070		No DMC11s
DZ11	160100	300	
GAP	160110		No more DZ11s

Next, a system with one DJ11, one DH11, one GT40, one KW11-W, and two DZ11s.

Option	Address	Vector	Comment
DJ11	160010	300	
GAP	160020		No more DJ11s
GAP	160030		DH11 must start on an address boundary that is a multiple of 20.
DH11	160040	310	
	160050		
GAP	160060		No more DH11s
GT40		320	GT40 address is not in the floating address space.
KW11-W		330	KW11-W address is not in the floating address space.
GAP	160070		No DQ11s
GAP	160100		No DU11s
GAP	160110		No DUP11s
GAP	160120		No LK11s
GAP	160130		No DMC11s
DZ11	160140	340	
DZ11	160150	350	
GAP	160160		No more DZ11s

3.2 REGISTER BIT ASSIGNMENTS

A comprehensive pictorial of all register bit assignments is shown in Figure 3-1. The four device registers (DR0, DR2, DR4, and DR6) are subdivided to form six unique registers. This subdivision is accomplished in DR2 and DR6 by assigning read-only (RO) or write-only (WO) status to each register. Since the reading and writing of DR2 and DR6 accesses two registers, PDP-11 processor instructions that perform a read-modify-write (DATIP) bus cycle cannot be used with DR2 or DR6. Also, DR2 permits only word instructions, but either byte or word instructions may be used with DR6. DR0 and DR4 have no programming constraints. In all register operations, the following applies: read-only bits are not affected by an attempt to write, and write-only and "not-used" bits appear as a binary 0 if a read operation is performed. Specific programming constraints for each register are discussed in the following paragraphs. A description of each bit function is presented in Tables 3-1 through 3-3.

3.2.1 Control and Status Register (CSR)

The control and status register (CSR) contains the states of flags and enable bits for scanning, processor interrupts, clearing, and maintenance. The 16-bit CSR has no programming constraints. The format is depicted in Figure 3-1, and bit functions are described in Table 3-1. Write-only and "not-used" bits are read as zeros by the Unibus, and read-only bits are not affected by write attempts.

	Register	15 (MSB)	14	13	12	11	10	09	08 (HIGH BYTE)	07 (LOW BYTE)	06	05	04	03	02	01	00 (LSB)
DR0	CONTROL & STATUS (CSR)	RO TRDY	RW TIE	RO SA	RW SAE	NOT USED	RO TLINE C	RO TLINE B	RO TLINE A	RO RDONE	RW RIE	RW MSE	RW CLR	RW MAINT	NOT USED	NOT USED	NOT USED
DR2	RECEIVER BUFFER (RBUF)	RO DATA VALID	RO OVRN	RO FRAM ERR	RO PAR ERR	NOT USED	RO RX LINE C	RO RX LINE B	RO RX LINE A	RO RBUF D7	RO RBUF D6	RO RBUF D5	RO RBUF D4	RO RBUF D3	RO RBUF D2	RO RBUF D1	RO RBUF D0
DR2	LINE PARAMETER (LPR)	NOT USED	NOT USED	NOT USED	WO RX ON	WO FREQ D	WO FREQ C	WO FREQ B	WO FREQ A	WO ODD PAR	WO PAR ENAB	WO STOP CODE	WO CHAR LGTH B	WO CHAR LGTH A	WO LINE C	WO LINE B	WO LINE A
DR4	TRANSMIT CONTROL (TCR)*	RW DTR 7	RW DTR 6	RW DTR 5	RW DTR 4	RW DTR 3	RW DTR 2	RW DTR 1	RW DTR 0	RW LINE ENAB 7	RW LINE ENAB 6	RW LINE ENAB 5	RW LINE ENAB 4	RW LINE ENAB 3	RW LINE ENAB 2	RW LINE ENAB 1	RW LINE ENAB 0
DR6	MODEM STATUS (MSR)*	RO CO 7	RO CO 6	RO CO 5	RO CO 4	RO CO 3	RO CO 2	RO CO 1	RO CO 0	RO RI 7	RO RI 6	RO RI 5	RO RI 4	RO RI 3	RO RI 2	RO RI 1	RO RI 0
DR6	TRANSMIT DATA (TDR)	WO BRK 7	WO BRK 6	WO BRK 5	WO BRK 4	WO BRK 3	WO BRK 2	WO BRK 1	WO BRK 0	WO TBUF 7	WO TBUF 6	WO TBUF 5	WO TBUF 4	WO TBUF 3	WO TBUF 2	WO TBUF 1	WO TBUF 0

*The high byte of the TCR (Data Terminal Ready) and the MSR are not used with the 20 mA options.

11-5313

Figure 3-1 Register Bit Assignments

Table 3-1 CSR Bit Functions

Bit	Title	Function
00–02	Not used	
03	Maintenance (MAINT)	A read/write bit that, when set, causes the serial output data from the transmitter to be fed back as serial input data to the receiver. All lines are turned around. Cleared by BUS INIT and CLR.
04	Clear (CLR)	A read/write bit that fires a one-shot to generate a 15 μs reset which clears the receiver silo, all UARTs, and the CSR. After a CLR is issued, the CSR and line parameters must be set again. CLR in progress is indicated by CLR = 1. Modem control registers are not affected, nor are bits 00 through 14 of RBUF.
05	Master Scan Enable	A read/write bit that activates the scanner to enable the receiver transmitter and silo. Cleared by CLR and BUS INIT.
06	Receiver Interrupt Enable	A read/write bit that enables the receiver interrupt. Cleared by CLR and BUS INIT.
07	Receiver Done (RDONE)	A read-only bit (hardware set) that generates RCV INT if bit 06 = 1 and bit 12 = 0. The bit clears when the RBUF is read and resets when another word reaches the output of the silo (RBUF). If bit 06 = 0, RDONE can be used as a flag to indicate that the silo contains a character. If bit 12 = 1, RDONE does not cause interrupts but otherwise acts the same.
08–10	Transmit Line A–C (TLINE)	When bit 15 = 1, these three read-only bits indicate the line that is ready to transmit a character. Bit 15 clears when the character is loaded into the transmit buffer, but sets again if another line is ready. A new line number could appear within a minimum of 1.9 μs. Bits 08–10 return to line 0 after a CLR or BUS INIT. These bits are meaningful only when bit 15 (TRDY) is true.
11	Not used	
12	Silo Alarm Enable (SAE)	A read/write bit that enables the silo alarm and prevents RDONE from causing interrupts. If bit 06 = 1, the SAE allows the SA (bit 13) to cause an interrupt after 16 entries in the silo. If bit 06 = 0, the SA can be used as a flag. The bit is cleared by CLR and BUS INIT.

Table 3-1 CSR Bit Functions (Cont)

Bit	Title	Function
13	Silo Alarm (SA)	A read-only bit set by the hardware after 16 characters enter the silo. It causes an interrupt if bit 06 = 1 and is cleared by CLR, BUS INIT, and reading the RBUF. When the silo flag occurs (SA = 1), the silo must be emptied because the flag will not be set again until 16 additional characters enter the silo.
14	Transmitter Interrupt Enable (TIE)	A read/write bit that allows an interrupt if bit 15 (TRDY) = 1.
15	Transmitter Ready (TRDY)	A read-only bit that is set by hardware when a line number is found that has its transmit buffer empty and its LINE ENAB bit set. It is cleared by CLR, BUS INIT, and by loading the TBUF register.

3.2.2 Receiver Buffer (RBUF)

The receiver buffer (RBUF) register contains the received character bits, with line identification, error status, and data validity flag. As one of two registers in DR2 (RBUF and LPR), RBUF is accessed when a read operation is performed (write operation accesses the LPR). The programming constraints for the RBUF register are as follows.

1. Byte instructions cannot be used.
2. It is a read-only register.
3. TST or BIT instructions cannot be used because they cause the loss of a character.
4. The register requires master scan enable (CSR, bit 05) to be set in order to be functional. When this bit is off, bits 00 to 14 of the RBUF become invalid regardless of the state of bit 15 (data valid) and the silo is held empty. The register format of RBUF is depicted in Figure 3-1 and bit functions are described in Table 3-2. Each reading of the RBUF register advances the silo and presents the next character to the program. Bits 00 through 14 do not go to zero after a CLR or BUS INIT; however, they become invalid and the silo is emptied. Bit 15 (data valid) does clear to zero. (See Table 3-2.)

Table 3-2 RBUF Bit Functions

Bit	Title	Function
00–07	Received Character	These bits contain the received character. If the selected code level is less than eight bits wide, the high-order bits are forced to zero.
08–10	Line Number	These bits present the line number on which the character was received.
11	Not used	
12	Parity Error	This bit indicates whether the received bit had a parity error. The parity bit is generated by hardware and does not appear in the RBUF word.
13	Framing Error	This bit indicates improper framing (stop bit not a mark) of the received character and can be used for break detection.
14	Overrun	This bit indicates receiver buffer overflow. The result is a received character which is replaced by another received character before storage in the silo. A character is lost but the received character put in the silo is valid.
15	Data Valid	This bit indicates that the character read from the silo (RBUF) is valid. The RBUF is read until the data valid bit = 0, indicating an invalid character and empty silo. Cleared by CLR and BUS INIT.

3.2.3 Line Parameter Register (LPR)

The line parameter register (LPR) is a 16-bit register that sets the parameters (character and stop code lengths, parity, speed, and receiver clock) for each line (Table 3-3). Bits 00–02 select the line for parameter loading. Line parameters for each line must be reloaded after a CLR (bit 04 of CSR) or BUS INIT operation. The programming constraints for the LPR are as follows.

1. It is a write-only register.
2. BIS or BIC instructions are not allowed.
3. Byte operations cannot be used.

Table 3-3 LPR Bit Functions

Bit	Title	Function
00–02	Line Number	These bits select the line for parameter loading
03–04	Character Length	These bits set the character length for the selected line. The parity bit is not part of the character length. **04 03** 0 0 5 bits 0 1 6 bits 1 0 7 bits 1 1 8 bits
05	Stop Code	This bit sets the stop code length (0 = 1-unit stop, 1 = 2-unit stop or 1.5-unit stop if a 5-level code is employed).
06	Parity	This bit selects the parity option (0 = no parity check, 1 = parity enabled on TRAN and RCV).
07	Odd Parity	This bit selects the kind of parity (0 = even parity select, 1 = odd parity select). Bit 06 must be set for this bit to have effect.
08–11	Speed Select	These bits select the TRAN and RCV speed for the line selected by bits 00–02. Refer to Table 3-4 for a list of available baud rates.
12	Receiver On	This bit must be set when loading parameters to activate the receiver clock. (Transmitter clock is always on.) A CLR or BUS INIT turns the receiver clock off.

3.2.4 Transmit Control Register (TCR)

The transmit control register contains 16 bits for the EIA options (M7819 module) and 8 bits for the 20 mA option (7814 module). The difference is that the data terminal ready (DTR) lines that make up the high byte (bits 08 through 15) of the TCR are not used by the 20 mA options because they do not have modem control capabilities.

The high byte (M7819 only) contains a read/write DTR bit for each line. This byte is cleared by BUS INIT only, not by CLR. When the high byte is not used (M7814 only), it reads back to the Unibus as all zeros. Attempts to write into it will have no effect. The low byte contains a read/write line enable bit for each line. A set bit allows transmission on the corresponding line. Paragraph 3.3.7 explains how to properly use this bit. This byte is cleared by CLR and BUS INIT.

3.2.5 Modem Status Register (MSR)

This is a 16-bit register used only with the EIA options (M7819 module). The 20 mA options (M7814 module) do not have modem control capabilities. When not used, this register reads all zeros to the Unibus.

The MSR consists of two bytes: the low byte (bits 00–07) and the high byte (bits 08–15). The low byte monitors the state of each line's ring indicator (RI) lead; the high byte monitors the state of each line's carrier (CO) lead. The MSR is the read-only portion of DR6 and has the following programming characteristics.

1. It is a read-only register.
2. CLR and BUS INIT have no effect.
3. Bit format is shown in Figure 3-1.

3.2.6 Transmit Data Register (TDR)

The TDR consists of two 8-bit bytes. The low byte is the transmit buffer (TBUF) and holds the character that is to be transmitted. The high byte is the break register with each line controlled by an individual bit. When a break bit is set, the line associated with that bit starts sending zeros immediately and continuously. The TDR is the write-only portion of DR6 and has the following programming characteristics.

1. It is a write-only register.
2. BIS or BIC instructions cannot be used.
3. For character lengths less than 8 bits, the character loaded into the TBUF must be right justified because the hardware forces the most significant bits to zero.
4. The break register has no effect when running in the maintenance mode (i.e., CSR bit 03 = 1).
5. It is cleared by CLR and BUS INIT.
6. Bit format is shown in Figure 3-1.

3.3 PROGRAMMING FEATURES

The DZ11 has several programming features that allow control of baud rate, character length, stop bits, parity, and interrupts. This section discusses the application of these controls to achieve the desired operating parameters.

3.3.1 Baud Rate

The selection of the desired transmission and reception speed is controlled by the conditions of bits 08 through 11 of the LPR. Table 3-4 depicts the required bit configuration for each operating speed. The baud rate for each line is the same for both the transmitter and receiver. The receiver clock is turned on and off by setting and clearing bit 12 in the LPR for the selected line.

Table 3-4 Baud Rate Selection Chart

Bits				
11	10	09	08	Baud Rate
0	0	0	0	50
0	0	0	1	75
0	0	1	0	110
0	0	1	1	134.5
0	1	0	0	150
0	1	0	1	300
0	1	1	0	600
0	1	1	1	1200
1	0	0	0	1800
1	0	0	1	2000
1	0	1	0	2400
1	0	1	1	3600
1	1	0	0	4800
1	1	0	1	7200
1	1	1	0	9600
1	1	1	1	Not used

3.3.2 Character Length

The selection of one of the four available character lengths is controlled by bits 03 and 04 of the LPR. The bit conditions for bits 04 and 03, respectively, are as follows: 00 (5-level), 01 (6-level), 10 (7-level), and 11 (8-level). For character lengths of 5, 6, and 7, the high-order bits are forced to zero.

3.3.3 Stop Bits

The length of the stop bits in a serial character string is determined by bit 05 of the LPR. If bit 05 is a zero, the stop length is one unit; bit 05 set to a one selects a 2-unit stop unless the 5-level character length (bits 03 and 04 at zero) is selected, in which case the stop bit length is 1.5 units.

3.3.4 Parity

The parity option is selected by bit 06 of the LPR. Parity is enabled on transmission and reception by setting bit 06 to a one. Bit 07 of the LPR allows selection of even or odd parity, and bit 06 must be set for bit 07 to be significant. The parity bit is generated and checked by hardware, and does not appear in the RBUF or TBUF. The parity error (bit 12, RBUF) flag is set when the received character has a parity error.

3.3.5 Interrupts

The receiver interrupt enable (RIE) and silo alarm enable (SAE) bits in the CSR control the circumstances upon which the DZ11 receiver interrupts the PDP-11 processor.

If RIE and SAE are both clear, the DZ11 never interrupts the PDP-11 processor. In this case, the program must periodically check for the availability of data in the silo and empty the silo when data is present. If the program operates off a clock, it should check for characters in the silo at least as often as the time it takes for the silo to fill, allowing a safety factor to cover processor response delays and time to empty the silo. The RDONE bit in the CSR will set when a character is available in the silo. The program can periodically check this bit with a TSTB or BIT instruction. When RDONE is set, the program should empty the silo.

If RIE is set and SAE is clear, the DZ11 will interrupt the PDP-11 processor to the DZ11 receiver vector address when RDONE is set, indicating the presence of a character at the bottom of the silo. The interrupt service routine can obtain the character by performing a MOV instruction from the RBUF. If the program then dismisses the interrupt, the DZ11 will interrupt when another character is available (which may be immediately if additional characters were placed in the silo while the interrupt was being serviced). Alternatively, the interrupt service routine may respond to the interrupt by emptying the silo before dismissing the interrupt.

If RIE and SAE are both set, the DZ11 will interrupt the PDP-11 processor to the DZ11 receiver vector when the silo alarm (SA) bit in the CSR is set. The SA bit will be set when 16 characters have been placed in the silo since the last time the program has accessed the RBUF. Accessing the RBUF will clear the SA bit and the associated counter. The program should follow the procedure described in Paragraph 3.3.6 to empty the silo completely in response to a silo alarm interrupt. This will ensure that any characters placed in the silo while it is being emptied are processed by the program.

NOTE

If the program processes only 16 entries in response to each silo alarm interrupt, characters coming in while interrupts are being processed will build up without being counted by the silo alarm circuit and the silo may eventually overflow without the alarm being issued.

If the silo alarm interrupt is used, the program will not be interrupted if fewer than 16 characters are received. In order to respond to short messages during periods of moderate activity, the PDP-11 program should periodically empty the silo. The scanning period will depend on the required responsiveness to received characters. While the program is emptying the silo, it should ensure that DZ11 receiver interrupts are inhibited. This should be done by raising the PDP-11 processor priority. The silo alarm interrupt feature can significantly reduce the PDP-11 processor overhead required by the DZ11 receiver by eliminating the need to enter and exit an interrupt service routine each time a character is received.

The transmitter interrupt enable bit (TIE) controls transmitter interrupts to the PDP-11 processor. If enabled, the DZ11 will interrupt the PDP-11 processor to the DZ11 transmitter interrupt vector when the transmitter ready (TRDY) bit in the CSR is set, indicating that the DZ11 is ready to accept a character to be transmitted.

3.3.6 Emptying the Silo

The program can empty the silo by repeatedly performing MOV instructions from the RBUF to temporary storage. Each MOV instruction will copy the bottom character in the silo so it will not be lost and will clear out the bottom of the silo, allowing the next character to move down for access by a subsequent MOV instruction. The program can determine when it has emptied the silo by testing the data valid bit in each word moved out of the RBUF. A zero value indicates that the silo has been emptied. The test can be performed conveniently by branching on the condition code following each MOV instruction. A TST or BIT instruction must not access the RBUF because these instructions will cause the next entry in the silo to move down without saving the current bottom character. Furthermore, following a MOV from the RBUF, the next character in the silo will not be available for at least 1 μs. Therefore, on fast CPUs, the program must use sufficient instructions or NOPs to ensure that successive MOVs from the RBUF are separated by a minimum of 1 μs. This will prevent a false indication of an empty silo.

3.3.7 Transmitting a Character

The program controls the DZ11 transmitter through five registers on the Unibus: the control and status register (CSR), the line parameter register (LPR), the line enable register, the transmitter buffer (TBUF), and the break register (BRK).

Following DZ11 initialization, the program must use the LPR to specify the speed and character format for each line to be used and must set the master scan enable (MSE) bit in the CSR. The program should set the transmitter interrupt enable (TIE) bit in the CSR if it wants the DZ11 transmitter to operate on a program interrupt basis.

The line enable register is used to enable and disable transmission on each line. One bit in this 8-bit register is associated with each line. The program can set and clear bits by using MOV, MOVB, BIS, BISB, BIC, and BICB instructions. (If word instructions are used, the line enable register and the DTR registers on M7819 modules are simultaneously accessed.)

The DZ11 transmitter is controlled by a scanner which is constantly looking for an enabled line (line enable bit set) which has an empty UART transmitter buffer. When the scanner finds such a line, it loads the number of the line into the 3-bit transmit line number (TLINE) field of the CSR and sets the TRDY bit, interrupting the PDP-11 processor if the TIE bit is set. The program can clear the TRDY bit by moving a character for the indicated line into the TBUF or by clearing the line enable bit. Clearing the TRDY bit frees the scanner to resume its search for lines needing service.

To initiate transmission on an idle line, the program should set the TCR bit for that line and wait for the scanner to request service on the line, as indicated by the scanner loading the number of the line into TLINE and setting TRDY. The program should then load the character to be transmitted into the TBUF by using a MOVB instruction. If interrupts are to be used, a convenient way of starting up a line is to set the TCR bit in the main program and let the normal transmitter interrupt routine load the character into the TBUF.

NOTE

The scanner may find a different line needing service before it finds the line being started up. This will occur if other lines request service before the scanner can find the line being started. The program must always check the TLINE field of the CSR when responding to TRDY to ensure it loads characters for the correct line. Assuming the program services lines as requested by the scanner, the scanner will eventually find the line being started. If several lines require service, the scanner will request service in priority order as determined by line number. Line 7 has the highest priority and line 0 the lowest.

To continue transmission on a line, the program should load the next character to be transmitted into the TBUF each time the scanner requests service for the line as indicated by TLINE and TRDY.

To terminate transmission on a line, the program loads the last character normally and waits for the scanner to request an additional character for the line. The program clears the line enable bit at this time instead of loading the TBUF.

The normal rest condition of the transmitted data lead for any line is the 1 state. The break register (BRK) is used to apply a continuous zero signal to the line. One bit in this 8-bit register is associated with each line. The line will remain in this condition as long as the bit remains set. The program should use a MOVB instruction to access the BRK register. If the program continues to load characters for a line after setting the break bit, transmitter operation will appear normal to the program despite the fact that no characters can be transmitted while the line is in the continuous zero sending state. The program may use this facility for sending precisely timed zero signals by setting the break bit and using transmit ready interrupts as a timer.

It should be remembered that each line in the DZ11 is double buffered. The program must not set the BRK bit too soon or the two data characters preceding the break may not be transmitted. The program must also ensure that the line returns to the 1 state at the end of the zero sending period before transmitting any additional data characters. The following procedure will accomplish this. When the scanner requests service the first time after the program has loaded the last data character, the program should load an all-zero character. When the scanner requests service the second time, the program should set the BRK bit for the line. At the end of the zero sending period, the program should load an all-zero character to be transmitted. When the scanner requests service, indicating this character has begun transmission, the program should clear the BRK bit and load the next data character.

3.3.8 Data Set Control

DZ11 models with EIA interfaces include data set control as a standard feature. The program may sense the state of the carrier and ring indicator signals from each data set and may control the state of the data terminal ready signal to each data set. The program uses three 8-bit registers to access the DZ11 data set control logic. One bit in each register is associated with each of the eight lines. There are no hardware interlocks between the data set control logic and the receiver and transmitter logic. Any required coordination should be done under program control.

The data terminal ready (DTR) register is a read/write register. Setting or clearing a bit in this register will turn the appropriate DTR signal on or off. The program may access this register with word or byte instructions. (If word instructions are used, the DTR and line enable registers will be simultaneously accessed.) The DTR register is cleared by the INIT signal on the Unibus but is not cleared if the program clears the DZ11 by setting the CLR bit of the CSR.

The carrier register (CAR) and ring register (RING) are read-only registers. The program can determine the current state of the carrier signal for a line by examining the appropriate bit of the CAR register. It can determine the current state of the ring signal by examining the appropriate bit of the ring register. The program can examine these registers separately by using MOVB or BITB instructions or can examine them as a single 16-bit register by using MOV or BIT instructions. The DZ11 data set control logic does not interrupt the PDP-11 processor when a carrier or ring signal changes state. The program should periodically sample these registers to determine the current status. Sampling at a high rate is not necessary.

3.4 PROGRAMMING EXAMPLES

The following six examples are sample programs for the DZ11 option. These examples are presented only to indicate how the DZ11 can be used.

Example 1 - Initializing the DZ11
The DZ11 is initialized by a power-up sequence, a reset instruction, or a device clear instruction.

Device Clearing the DZ11

```
001000  012737  START:  MOV #20, DZCSR   ;Set bit 4 in the
001002  000020                           ;DZ11 control and
001004  160100                           ;status register.
001006  032737  1$:     BIT #20, DZCSR   ;Test bit 4.
001010  000020
001012  160100
001014  001374          BNE 1$           ;If bit 4 is still
                                         ;set, the branch
                                         ;condition is true
                                         ;and the device clear
                                         ;function is still in
                                         ;progress.
001016  000000          HALT             ;The device clear
                                         ;function is complete
                                         ;and the DZ11 has been
                                         ;cleared.
```

DZCSR = Control and Status Register Address = 160100.

Example 2 - Transmit Binary Count Pattern on One Line

```
001000  012737  START:  MOV #20, DZCSR   ;Set bit 4 in the DZ11
001002  000020                           ;control and status
                                         ;register.
001004  160100
001006  032737  1$:     BIT #20, DZCSR   ;Test bit 4.
001010  000020
001012  160100
001014  001374          BNE 1$           ;If bit 4 is still set,
                                         ;the branch condition
                                         ;is true and the device
                                         ;clear function is still
                                         ;in progress.
001016  012737          MOV #n, DZLPR    ;Load the parameters
001020  001070                           ;for line 0: 8-bit
001022  160102                           ;character; 2 stop bits;
                                         ;110 baud
001024  012737          MOV #1, DZTCR    ;Enable line 0
                                         ;transmitter.
```

```
001026    000001
001030    160104
001032    012737              MOV #m, DZCSR       ;Set scanner enable bit
001034    000040                                  ;5 in the control and
001036    160100                                  ;status register.
001040    005000              CLR R0              ;Set binary count
                                                  ;pattern to zero.
001042    005737    2$:       TST DZCSR           ;Test the transmitter
001044    160100                                  ;ready flag (bit 15).
001046    100375              BPL 2$              ;If branch condition
                                                  ;is false, continue;
                                                  ;otherwise test again.
001050    110037              MOVB R0, DZTDR      ;Load character to be
001052    160106                                  ;transmitted.
001054    105200              INCB R0             ;Increment binary count.
001056    100371              BPL 2$              ;If branch condition is
                                                  ;false, the binary count
                                                  ;pattern is complete.
001060    000000              HALT
```

R0 = Register 0 = Binary Count Pattern
DZCSR = DZ11 Control and Status Register Address = 160100
DZLPR = DZ11 Line Parameter Register Address = 160102
DZTCR = DZ11 Transmit Control Register Address = 160104
DZTDR = DZ11 Transmit Data Register Address = 160106

Example 3 – Transmit a Binary Count in Maintenance Loopback Mode, with the Receiver "On" in the Interrupt Mode

Output Received Data to Console

```
001200    005000              CLR R0              ;Set binary count
                                                  ;to zero.
001202    012701              MOV 1400, R1        ;Set R1 to first
001204    001400                                  ;address of data
                                                  ;buffer.
001206    012706              MOV #SP, R6         ;Initialize stack
001210    001100                                  ;pointer.
001212    012737              MOV #INT, RVEC      ;Set DZ11 vector
001214    001304                                  ;address to start of
001216    000300                                  ;receiver interrupt
                                                  ;routine.
001220    005037              CLR (RVEC+2)        ;Set up processor
001222    000302                                  ;status word for DZ11
                                                  ;receiver interrupt.
001224    012737              MOV #20, DZCSR      ;Set bit 4 in the
001226    000020                                  ;DZ11 control and
                                                  ;status register.
```

```
001230  160100
001232  032737   1$:   BIT #20 DZCSR       ;Test bit 4.
001234  000020
001236  160100
001240  001374         BNE 1$              ;If bit 4 is still
                                           ;set, the branch
                                           ;condition is true
                                           ;and the device clear
                                           ;function is still in
                                           ;progress.
001242  012737         MOV #PAR, DZLPR     ;Load the parameters
001244  011070                             ;for line 0: 8-bit
001246  160102                             ;character; 2 stop bits;
                                           ;110 baud; no
                                           ;parity; receiver on.
001250  012737         MOV #1, DZTCR       ;Enable line 0
                                           ;transmitter.
001252  000001
001254  160104
001256  012737         MOV #150, DZCSR     ;Turn scanner on,
001260  000150                             ;enable receiver
001262  160100                             ;interrupts, and loop
                                           ;lines back on themselves.
001264  005737   2$:   TST DZCSR           ;Test the transmitter
001266  160100                             ;ready flag.
001270  100375         BPL 2$              ;If branch condition is
                                           ;false, continue;
                                           ;otherwise test again.
001272  110037         MOVB R0, DZTBUF     ;Load character to be
001274  160106                             ;transmitted.
001276  105200         INCB R0             ;Increment binary count.
001300  001371         BNE 2$              ;If branch condition is
                                           ;false, the binary count
                                           ;pattern is complete.
001302  000777         BR.                 ;Wait for last character
                                           ;transmitted to be
                                           ;received.
```

Receiver Interrupt Service Routine

```
001304   013711           MOV DZRBUF,(R1)    ;Store received
001306   160102                              ;character in memory
                                             ;table.
001310   022721           CMP #100377,       ;Check for last
001312   100377           (R1)+              ;character.
001314   001401           BEQ .+2            ;Branch condition is
                                             ;true when last
                                             ;transmitted character
                                             ;is received.
001316   000002           RTI                ;Exit routine.
001320   012701           MOV #1400, R1      ;Initialize pointer
001322   001400                              ;to start of received
                                             ;data buffer in memory.
001324   105737   3$:     TSTB TPS           ;Test to see if console
001326   177564                              ;is ready.
001330   100375           BPL 3$             ;Wait, and test again.
                                             ;If condition is met,
001332   111137           MOVB (R1), TPB     ;transfer character
001334   177566                              ;to console.
001336   022721           CMP #100377,       ;Check for last
001340   100377           (R1)+              ;character.
001342   001370           BNE 3$             ;Not finished if
                                             ;condition is true.
001344   000000           HALT               ;finished.
```

RVEC = DZ11 Receiver Interrupt Vector Address
DZCSR = DZ11 Control and Status Word Address
DZLPR = DZ11 Line Parameter Register (Write-Only) Address
DZTCR = DZ11 Transmit Control Register Address
DZTBUF = DZ11 Transmit Buffer Address
DZRBUF = DZ11 Receiver Buffer Address (Read-Only Register)
TPS = Teletype® Punch Status Register Address
TPB = Teletype Punch Data Register Address

®Teletype is a registered trademark of Teletype Corporation.

Example 4 - Transmit and receive in Maintenance Mode on a Single Line
The switch register bits (SWR00–SWR07) hold the desired data pattern (character).

```
001000   012737   START:   MOV #LINE, DZTCR    ;Select the line for
001002   000002                                ;transmitting on.
                                               ;Choose one of eight.
001004   160104                                ;Line #1 selected.
001006   012737            MOV #PAR, DZLPR     ;Select desired line
001010   017471                                ;parameters for
                                               ;transmitting line
001012   160102                                ;and turn on receiver
                                               ;for that line.
                                               ;8-level code, 2 stop
                                               ;bits, and no parity
                                               ;selected.
                                               ;19.2K baud selected
                                               ;Note: 19.2K baud is
                                               ;not used by the
                                               ;customer but can be
                                               ;used for diagnostic
                                               ;purposes to speed up
                                               ;the transmit-receive
                                               ;loop to make it easier
                                               ;to scope.
001014   012737            MOV #N, DZCSR       ;Start scanner and set
001016   000050                                ;maintenance bit 3.
001020   160100
001022   005737   Test 1:  TST DZCSR           ;Test for bit 15
001024   160100                                ;(transmitter ready).
001026   100375            BPL Test 2          ;If the branch condition
                                               ;is false, the transmitter
                                               ;is ready; if true, go
                                               ;back and test again.
001030   113737            MOVB SWR,           ;Load the transmit
001032   177570            DZTBUFF             ;character from the
001034   160106                                ;switch register.
001036   000240            NOP                 ;No operation. This
                                               ;location can be changed
                                               ;to a branch instruction
                                               ;if only test 1 is
                                               ;desired (replace 000240
                                               ;with 000771).
001040   012701            MOV #DEL, R1        ;Delay equals a
         177670                                ;constant that will
                                               ;allow enough time for
                                               ;the receiver done
                                               ;flag to set before
                                               ;recycling the test.
                                               ;The value will change
                                               ;with baud rate and
                                               ;processor. The
                                               ;constant given is
                                               ;good for 19.2K baud
                                               ;on a PDP-11/05.
```

```
001042   105737   Test 2:   TSTB DZCSR        ;Test bit 2 (receiver
001044   160100                               ;done flag).
001046   100402             BMI 1$            ;When the branch
                                              ;condition is true,
                                              ;the receiver done
                                              ;flag is set.
001050   005201             INC R1            ;Increment delay.
001052   001373             BNE TEST 2        ;If the branch
                                              ;condition is true, the
                                              ;delay is not finished.
001054   013700   1$:       MOV DZRBUF, R0    ;Read the DZ11
001056   160102                               ;receiver buffer to
                                              ;register 0.
001060   000760             BR TEST 1         ;Loop back and
                                              ;test again.
```

Example 5 – Transmit and Receive on a Single Line Using Silo Alarm in Maintenance Mode

```
001200   012706             MOV #1100, R6     ;Initialize stack
001202   001100                               ;pointer.
001204   012737             MOV #3$, TVEC     ;Initialize transmitter
001206   001274                               ;vector address.
001210   000304
001212   005037             CLR TVEC+2        ;Initialize transmitter
001214   000306                               ;vector processor status
                                              ;word.
001216   012700             MOV #DBUF, R0     ;Set first address of
001220   001304                               ;input data table
                                              ;into R0.
001222   012737             MOV #1, DZTCR     ;Enable line 0
                                              ;transmitter.
001224   000001
001226   160104
001230   012737             MOV #17470,       ;Set up line parameters
001232   017470             DZLPR             ;and turn on the receiver
001234   160102                               ;clock for line 0.
001236   012737             MOV #50050,       ;Enable transmitter
001240   050050             DZCSR             ;interrupt and silo
001242   160100                               ;alarm. Turn on
                                              ;scanner and maintenance
                                              ;mode.
001244   032737   1$:       BIT #20000,       ;Test for silo alarm
001246   020000             DZCSR
001250   160100
001252   001774             BEQ 1$            ;Loop until silo alarm
                                              ;flag sets.
001254   013720   2$:       MOV DZRBUF,       ;Read DZ11 silo
001256   160102             (R0)+             ;receiver buffer output.
001260   000240             NOP               ;Delay to allow next
001262   000240             NOP               ;word in silo to filter
                                              ;down to the silo
                                              ;output.
```

```
001264    100773              BMI 2$                  ;Data valid set says
                                                      ;that word is good,
                                                      ;go back for more.
001266    012700              MOV #DBUF, R0           ;Silo has been emptied.
001270    001304                                      ;Reinitialize data
                                                      ;table address pointer.
001272    000764              BR 1$                   ;Do it again.
```

Transmitter Interrupt Service Routine

```
001274    112737     3$       MOVB DAT, DZTBUF        ;Transmit
001276    000252                                      ;character 252
001300    160106
001302    000002              RTI
```

Data Table

```
1304    100252    ;Word 1
1306    100252

 .        .         .
          .         .
 .        .         .
 .        .         .
 .        .         .
 .        .         .

1340    100252    ;Word 16
1342    000252    ;Data valid
                  ;not set
                  ;character is
                  ;invalid
```

NOTE

It is possible to get more than 16 words because they are being put into the silo simultaneously with the reading of the silo.

Example 6 – Echo Test on a Single Line (Transmit Received Data)

```
001000    012737    START    MOV #PAR, DZLPR       ;Load line parameters
                                                   ;for line being used.
001002    011073                                   ;Line 3, 8-bit
001004    160102                                   ;character, 2 stop
                                                   ;bits, no parity,
                                                   ;110 baud, and receiver
                                                   ;clock on.
001006    012737             MOV #LINE, DZTCR      ;Turn line 3
001010    000010                                   ;transmitter on.
001012    160104
001014    012737             MOV #n, DZCSR         ;Turn scanner on
001016    000040                                   ;(set CSR-5)
001020    160100
001022    105737    1$:      TSTB DZCSR            ;Test (bit 7) for
001024    160100                                   ;RDONE
001026    100375             BPL 1$                ;If bit 7 is not set,
                                                   ;go back and test again.
001030    005737    2$:      TST DZCSR             ;Test (bit 15) for
001032    160100                                   ;TRDY
001034    100375             BPL 2$                ;If bit 15 is not set
                                                   ;go back and test again.
001036    013700             MOV RBUF, R0          ;Read received data
001040    160102                                   ;word into R0
001042    110037             MOVB R0, DZTDR        ;Load character
001044    160106                                   ;into DZ11 TBUF
                                                   ;register for
                                                   ;transmitting.
001046    000765             BR 1$                 ;Repeat.
```